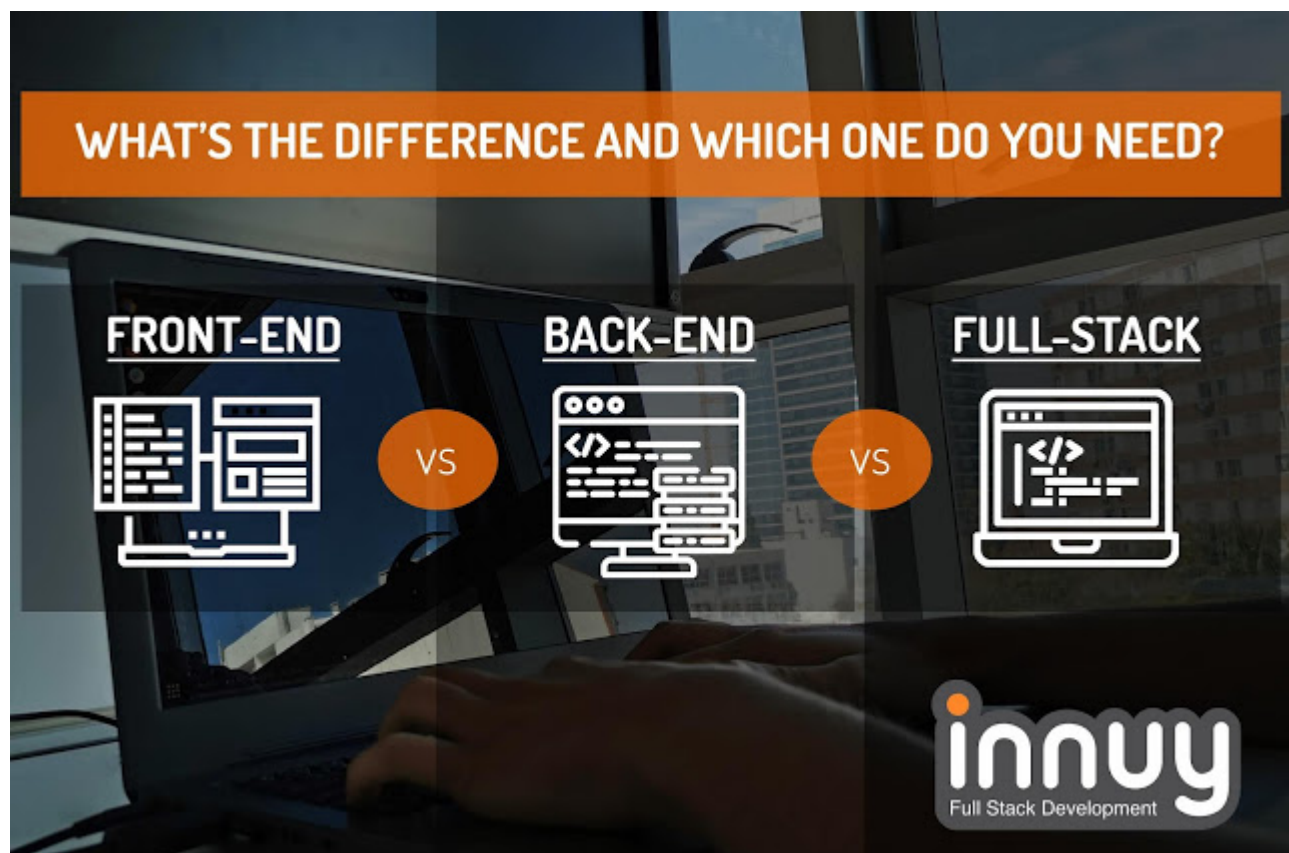


Front End vs. Back End vs. Full Stack: What's the Difference and Which One Do You Need?

on September 23, 2019.



If you've spent any time researching options for upgrading your digital presence or launching a new Web app, you've probably figured out the world of Web development has a lot of confusing jargon and tech-speak all its own.

While defining all the latest acronyms and tech-babble would take a Webster's Dictionary-sized volume, we can simplify a few basic terms pretty fundamental to Web apps to help you find a development company with the right skills for your needs.

Three phrases you've probably come across are front end, back end, and full-stack. Let's start by looking at the major differences between front end vs. back end vs. full stack when it comes to how a Web app works and also explore what each type of developer does.

Front End



Basics

The front end of a Web application is also called “client-side,” meaning it encompasses everything the browser (or client) displays or the user sees and interacts with (buttons, forms, links, etc.).

Put another way, the front end of a Web site can be defined as the way it looks, how it functions, what it says, and how a visitor interacts with its components within the browser window. When a site visitor clicks a colored button, watches a video, or pauses to read a paragraph of text, all of those features of the site have been rendered and delivered to the user at the front end.

Structure

The “structure” of the front end could be explained simply as the content, graphics, and architecture (or organization) of the Web site. However, developers also include the code behind what the user sees when referring to the “front end” portion of a Web site because there is a definite distinction there from code typically used on the back end.

Languages

The coding languages that help power the front end of a site include HTML for page structure and organization, CSS for look and feel, and Javascript for interactivity. Javascript, in turn, includes various libraries and frameworks (React, Angular, etc.) that help provide different flavors of interactivity and can assist with connecting the front end with the back end.

So, in turn, a front end developer is one who works closely with graphic designers and writers and focuses more on coding the visuals and content of a site than, say, how data might be stored or where.

Back End



Basics

Conversely, the back end of a Web site is described as “server-side” because it typically includes all the data-driven functionality that runs on a Web server in the background of a site, which users never see.

The back end of a site usually has three functions: data storage, connectivity, and communication. It’s the engine under the “hood” of a site, driving how a Web site interprets and responds to user interactions.

Structure

The structure of the back end of a Web site can range from incredibly simple to extremely complex, depending on user volume and site functionality. But every back end typically follows the same basic structure. Some type of database stores all the data used to power the Web site and make it useful to a visitor. A set of protocols and networking apps connect the browser to the server and power interactions between the two. And then a server and a set of server-side applications handle the site user’s instructions and inputs and communicate responses back to them.

Languages

The variety of languages used for the back end of a site or app is enormous and highly dependent on developer preference and skill as well as the site’s purpose, size, and functionality. A few of the most common back end languages include Python and PHP for server-side applications, SQL Server for data storage, and Linux, which is a type of server.

Javascript can also be used on the back end for simple functions, but the larger and more complex a site or mobile app is the more likely a backend-specific programming language will be necessary.

Back end developers, then, spend a lot of time thinking about the business needs of a site or app and how best to translate those needs into the server-side code to power that functionality. They worry way less about how a site looks to a user, preferring to leave that to the front end folks.

Full Stack



Basics

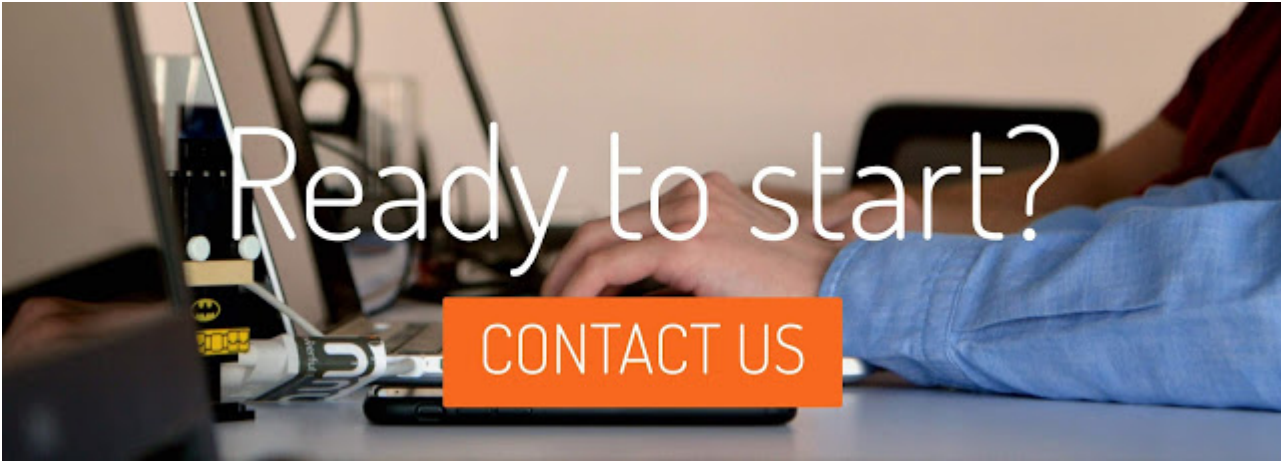
“Full-stack,” then, means, essentially, the full package – front end plus back end. The term “stack” in this case refers to all of the layers of a Web site: look and feel, interactivity, structure, data storage and manipulation, connectivity, and response.

Structure/Languages

The unique combination of structure and languages that make up the technology stack of a Web site depends completely on its purpose, traffic, and functionality needs and choosing [exactly the stack](#) you need is an important decision.

Full-stack developers are jacks-of-all-trades, with expertise and talent in both areas – front and back end. While they may have a focus or preference for one or the other, they deeply understand the technology and flow of the entire cycle of a Web site, from back to front and back again.

When hiring a developer or company to take on your project, it’s important to find one with the right focus for your particular needs. [Innuys team](#) of highly skilled full-stack developers can help you figure out the best solution and provide the tech knowledge to get it done.



Ready to start?

CONTACT US